Ferramentas e Ambientes para Objetos de Aprendizagem

André Santanchè, Andrea Lago, Patrícia Dourado, Pedro Ferreira Filho

NUPPEAD – Núcleo de Pesquisa e Projetos em Educação a Distância
NUPERC – Núcleo de Pesquisa Interdepartamental em Redes de Computadores
Universidade Salvador – UNIFACS
Av. Cardeal da Silva, 747 – Salvador – BA – Brasil

santanche@unifacs.br, brdea@yahoo.com.br,
patricia.almeida@si.unifacs.br, pedrff@gmail.com

Resumo. *Este texto conduz o leitor a uma percepção prática da noção de Objeto de Aprendizagem (OA). Para isso, são detalhados os principais padrões abertos que têm sido construídos em torno dessa noção, bem como são apresentados: estudos de caso; ferramentas para a criação/edição de OAs; a implementação de OAs em um Sistema para Gerenciamento de Cursos – o Moodle.*

Abstract. *This text conducts the reader to a practical perception of the Learning Object (LO) notion. For this purpose, we detail the main open standards built around this notion and we also present: case studies; tools for creating/editing LOs; an implementation of LOs in a Learning Management System – the Moodle.*

Apresentação

O conteúdo deste texto se baseia em uma produção nossa anterior [Santanchè 2007]. Seu papel é atuar de forma complementar ao anterior – que buscou principalmente sistematizar o conceito de OA, descrevendo cinco características distintivas do mesmo – , bem como apresentar padrões abertos relacionados a OAs. Neste texto se retomam alguns tópicos do texto anterior, complementando-o nos seguintes aspectos: é proposto um novo exemplo prático baseado em um WebQuest; a seção de padrões foi ampliada de modo a detalhar aqueles padrões relacionados com a comunicação entre o OA e o Sistema de Gerenciamento de Cursos (SGC); são apresentados dois sistemas de autoria de conteúdo educacional que exportam seu produto para padrões abertos de OA.

1. Introdução

A impressão causada pelo primeiro contato com a noção de Objeto de Aprendizagem (OA) traz a ambigüidade de uma idéia simultaneamente cativante e vaga. Cativa a idéia de poder criar estas "coisas", os objetos, os quais podem ser compartilhados e aplicados em atividades de ensino/aprendizagem. Objeto nos transmite a idéia de uma entidade que pode ser percebida e manipulada individualmente. Somos induzidos a imaginar repositórios, capazes de armazenar e classificar centenas desses objetos, de modo que possam posteriormente ser selecionados e reusados em contextos diversos.

Este texto é concedido sob uma Licença Creative Commons. Sob as seguintes condições: Atribuição, Uso Não-Comercial e Compartilhamento pela mesma Licença. Para maiores informações veja licença completa no site creativecommons.org.

Em contrapartida, uma reflexão mais aprofundada sobre o tema transmite a fragilidade de uma noção vaga. O que é exatamente um OA? Seria um software educacional completo, uma animação que é executada em um navegador Web, uma imagem? A tentativa de uma definição precisa comporta o risco dos extremos. De um lado, somos tentados a considerar tudo um OA e, neste caso, o termo perde o sentido. Como observam Sosteric and Hesemeier (2002), um termo que serve para definir tudo não define nada. Por outro lado, corremos o risco de criar definições extremamente restritivas, definindo um OA apenas como: um software executável, uma simulação, uma animação etc., e deixando de fora um rico conjunto de artefatos que podem igualmente ter um papel importante na construção de atividades de ensino-aprendizagem.

A caracterização de um OA deve considerar também sua natureza, que resulta da intersecção de dois domínios: Educação e Informática. Isto significa que não é suficiente definir um OA simplesmente do ponto de vista técnico da Informática, sem considerar os pressupostos pedagógicos da sua construção.

Este texto se propõe a contribuir na caracterização de um OA através de uma abordagem prática, onde são apresentados exemplos e ferramentas para a sua construção. Ele está organizado da seguinte maneira: na Seção 2, apresentamos exemplos de OAs e suas aplicações e sintetizamos um conjunto de características que, a nosso ver, são imprescindíveis na caracterização de um OA. Na Seção 3, detalhamos padrões abertos relacionados com OAs. Na Seção 4, apresentamos ferramentas para a construção e uso de OAs. Finalmente, na Seção 5, sumarizamos as considerações finais e perspectivas futuras.

2. Objetos de Aprendizagem

Para tornar mais concreta a apresentação do conteúdo a seguir, optamos por utilizar um conjunto de exemplos que ilustrarão cada um dos conceitos apresentados. Na próxima subseção serão apresentados exemplos do que consideramos OA. Nas subseções e seções subseqüentes faremos referência e analisaremos estes exemplos.

2.1. Exemplos de Objetos de Aprendizagem

Objeto de Aprendizagem 1 – Animação da Fagocitose

Objetos de aprendizagem podem ser de vários tamanhos e complexidade. Iniciaremos trabalhando com um exemplo bastante simples. Trata-se de uma seqüência de imagens que apresenta etapas de uma célula englobando outro corpo, através de um movimento conhecido como fagocitose, conforme ilustrado na Figura 1. Tal seqüência está armazenada em um único arquivo de imagem.

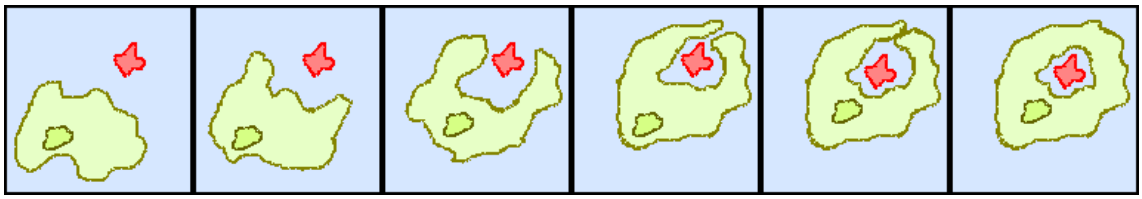
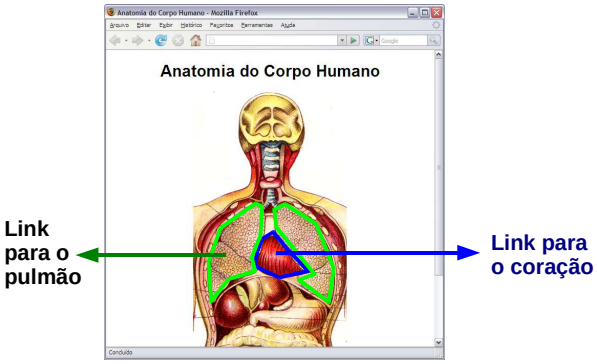


Figura 1. Exemplo de OA – seqüência de imagens ilustrando uma fagocitose.

Objeto de Aprendizagem 2 – Corpo Humano

Enquanto o exemplo anterior apresentou um objeto mais simples, que usualmente fará parte de produtos maiores, este segundo exemplo vai além, partindo para um objeto mais complexo que contém uma aplicação completa. Este exemplo foi originalmente apresentado em Dourado et al. (2007).

A produção do objeto abaixo pode ser feita por um professor de ciências ou biologia para os seus alunos, ou com os seus alunos. Trata-se de um material educativo sobre a anatomia do corpo humano, concebido para ser executado em um navegador Web. A Figura 2 apresenta a página inicial deste OA no momento em que ele é apresentado em um navegador Web; consiste em um conjunto de páginas HTML interligadas por links. Como está indicado na figura, a página principal utiliza um recurso do HTML que associa links a regiões geométricas da página.



Página Principal
(index.html)

Figura 2. Exemplo de OA para exploração interativa da anatomia humana.

O aluno pode clicar em partes do corpo humano e, através de links mapeados na imagem, será conduzido a outras páginas que contêm detalhes sobre o respectivo órgão. Para fins de simplificação, apenas uma parte do objeto, detalhando o coração e o pulmão, foi incluída no exemplo apresentado.

A Figura 3 apresenta um diagrama de dependências entre os arquivos que compõem o objeto. Pode-se observar a página principal no topo que: utiliza uma imagem da anatomia humana, define dois links para outras duas páginas com detalhes sobre o coração e o pulmão, e utiliza uma folha de estilo. A mesma interpretação pode ser usada para os demais arquivos.

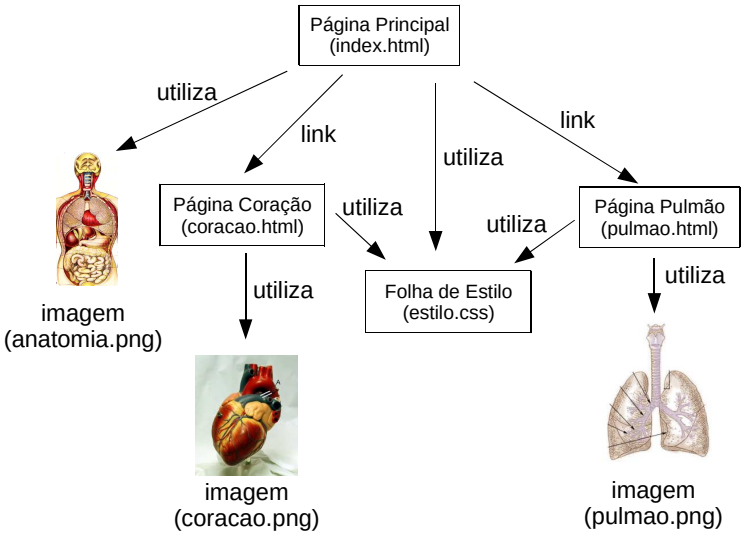


Figura 3. Arquivos que fazem parte do OA: anatomia e suas relações.

Objeto de Aprendizagem 3 – WebQuest

Este terceiro exemplo tem uma complexidade equivalente à do exemplo anterior e ilustra uma situação típica para aplicação de um objeto complexo: um WebQuest.

Um WebQuest consiste em um formato para o desenvolvimento de uma investigação orientada, na qual a maior parte das informações que o estudante trabalha vem da Web. Ele foi proposto por Bernie Dodge em 1995 [Dodge 1995] [Dodge 2001]. Mais detalhes sobre WebQuest podem ser encontrados no site oficial (em inglês) http://webquest.org ou no site em português da Escola do Futuro da USP http://www.webquest.futuro.usp.br.

Conforme ilustrado na Figura 4, um WebQuest usualmente é composto de um conjunto de páginas interligadas. Cada página corresponde a uma etapa do WebQuest: Introdução, Tarefa, Processo, Avaliação e Conclusão.

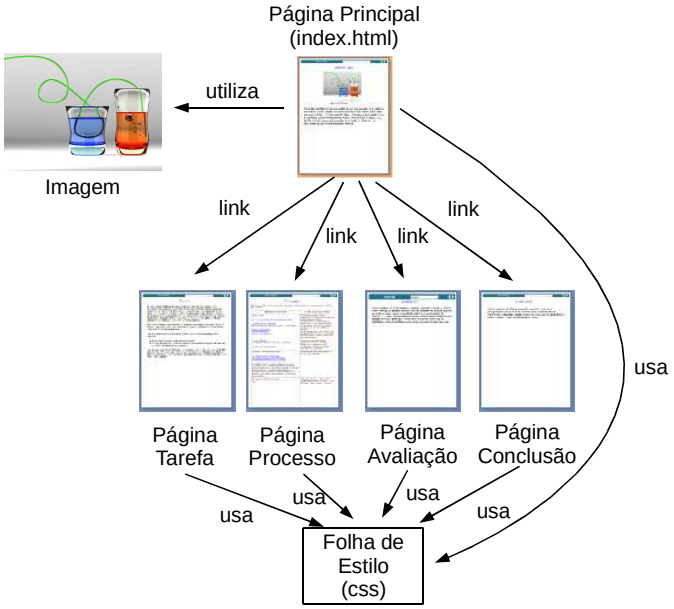


Figura 4. Diagrama da estrutura de um WebQuest na forma de OA.

Este WebQuest está sendo usado por nós em uma disciplina implantada no ambiente Moodle. Conforme será explanado adiante, representar o WebQuest desta forma permite empacotá-lo, arquivá-lo e reusá-lo como um pacote único. Além disso, ele é facilmente instalado no Moodle, que pode acompanhar o desenvolvimento do estudante dentro do WebQuest.

2.2. Características de um Objeto de Aprendizagem

A perspectiva do que representa software em geral - e em especial software educacional -, sofreu grandes modificações nos últimos anos. Gradativamente nos deslocamos de uma antiga noção de software fortemente amarrada àquela de código executável, para uma riqueza de possibilidades que inclui páginas Web, animações, produções multimídia. O código executável - que antigamente era produzido por alguns iniciados em computação - amplia suas possibilidades para múltiplas expressões de autores de todos os domínios. Hoje nos encontramos diante de um imenso repositório (a Web), compartilhado por alguns milhões de autores espalhados por todo o mundo. E daí surge a questão: Por que não utilizar toda essa riqueza de materiais já construídos? Por que sempre iniciar um novo material partindo do zero? Diante de um tempo cada vez mais escasso, da limitação de recursos e da encantadora diversidade de idéias e culturas expressas na forma de artefatos digitais, a prática do reuso se torna urgente.

Como veremos na próxima seção, tal necessidade se tornou premente não apenas na comunidade da Educação, dando origem a iniciativas com propostas e características equivalentes em diferentes domínios.

Uma das dificuldades em encontrar uma definição precisa para OA está relacionada com a origem do termo. É difícil dizer com precisão qual a sua origem. As definições encontradas não são equivalentes e os primeiros autores a usar o termo não se referenciam, dando a impressão de que estavam adotando a terminologia para contextos muito específicos, sem a intenção de generalizar aquele. Alguns autores [Wiley 2000, von Brevern 2004] relacionam e confrontam diferentes abordagens e definições para OA, mas não nos parece possível encontrar qualquer indício de unicidade. Na nossa edição anterior deste minicurso [Santanchè et al. 2007] explanamos com detalhes cinco características que consideramos essenciais para o OA, as quais resumimos aqui:

Unidade de reuso – Nosso ponto de vista é que a noção de OA está intimamente ligada à questão do reuso. Além disso, a capacidade de se adaptar é essencial para aumentar a possibilidade de reuso do objeto.

Digital – Ainda que a definição mais popular de OA inclua "qualquer entidade, digital ou não digital" [IEEE L.T.S.C., 2000], parece-nos sem sentido considerarmos as entidades não-digitais. Quando lidamos com computadores, tratamos sempre de entidade digitais e, por este motivo, praticamente todas as aplicações de OA encontradas lidam com entidades digitais [Sosteric e Hesemeier 2002].

Mais do que código executável – Uma pergunta recorrente de quem inicia na área de OA é se eles obrigatoriamente são programas de computador, ou seja, unidades feitas de código executável em alguma linguagem de programação. Nenhuma das definições originais – das mais antigas às mais populares – restringe os OAs a programas de computador [L'Allier 1997, IEEE L.T.S.C. 2000, Wiley 2000].

Aderência a padrões abertos – Para que seja possível o reuso de qualquer artefato digital é necessário algum tipo de padrão aberto, já que o reuso prevê um novo contexto que não aquele onde o artefato foi concebido.

Voltados à aprendizagem – Esta é uma característica derivada diretamente do nome. Por desnecessário que pareça destacá-la, ela ressalta que um OA não se resume a um conjunto de padrões técnicos, mas deve considerar principalmente os aspectos pedagógicos.

3. Padrões Abertos para Objetos de Aprendizagem

O principal segmento de pesquisas de padrões abertos relacionados com OAs tem suas origens a partir dos trabalhos do IEEE *Learning Technology Standardization Committee* (LTSC) (`http://ieeeltsc.org`). Este comitê é uma iniciativa para o desenvolvimento de tecnologias educacionais baseadas em padrões abertos de interoperabilidade. Especificamente o grupo de trabalho 12 (WG12), incumbido de definir do padrão de metadados, foi o responsável pela definição mais popular do termo *Learning Object* ou Objeto de Aprendizagem, em sua especificação para o LOM – *Learning Object Metadata* [IEEE L.T.S.C. 2000, IEEE L.T.S.C. 2002]. Associados a este padrão inicial, surgiram diversos padrões relacionados para:

- Metadados
- Empacotamento
- Interfaces e comunicação
- Seqüenciamento
- Integração dos demais padrões

Nas subseções a seguir analisaremos cada um destes padrões.

3.1. Metadados

Com uma vasta e crescente quantidade de OAs disponíveis na Web, a questão passa a ser como classificar e documentar tais objetos, de maneira que se possam projetar ferramentas aptas a encontrá-los e utilizá-los.

<u>Learning Object Metadata</u>

Com este objetivo, o LTSC desenvolveu o padrão LOM [IEEE L.T.S.C. 2002]. O padrão IEEE LOM deriva de trabalhos anteriores desenvolvidos na fundação européia ARIADNE (`http://www.ariadne-eu.org`), e pelo IMS Global Learning Consortium (`http://www.imsglobal.org`) [Nilsson et al. 2006].

No padrão LOM, os OAs são descritos por meio de um conjunto de propriedades, às quais são dados valores que caracterizam o objeto. O padrão define um esquema que relaciona hierarquicamente as propriedades aceitas, o domínio dos valores que elas podem receber, bem como sua cardinalidade. Este esquema, denominado *BaseScheme* (esquema básico), reúne as propriedades consideradas fundamentais para a descrição dos objetos. Ele pode ser estendido para outros esquemas derivados, de acordo com necessidades específicas.

A estrutura geral do *BaseScheme* é dividida segundo grupos de propriedades:

General	Dados de identificação, chaves de acesso, delimitação do domínio, descrição do conteúdo e da estrutura.
Life Cycle	Dados para controle e documentação do ciclo de vida do documento.
Meta-MetaData	Referências à origem e estrutura dos metadados.
Technical	Dados técnicos, tais como: formato, tamanho, requisitos de sistema operacional, duração, etc.
Educational	Elementos de descrição pedagógica do recurso, tais como: abordagem, nível de interatividade, pré-requisitos, objetivo educacional, grau de dificuldade, tempo esperado de trabalho, etc.
Rights	Dados referentes às condições de uso do produto e, eventualmente, valores a serem pagos pelo uso do recurso.
Relation	Características que relacionam o objeto de aprendizagem com outros.
Annotation	Comentários referentes ao uso educacional do produto.
Classification	Referência que determina onde o recurso será colocado dentro de um sistema de classificação específico.

Recentemente, o IEEE LTSC mapeou o esquema do LOM para XML [IEEE L.T.S.C. 2005], e o trabalho foi baseado em um padrão anterior encabeçado pelo consórcio IMS [Thropp and McKell 2001]. Para ilustrar como o LOM é criado e usado, apresentaremos um exemplo prático de elaboração de alguns metadados relacionados ao Exemplo 3 de OA (WebQuest). Para fazer isso será utilizada a ferramenta Reload Editor.

Ferramentas do projeto RELOAD

`http://www.reload.ac.uk`

O projeto RELOAD (*Reusable eLearning Object Authoring & Delivery*) tem como meta desenvolver ferramentas que oferecem suporte às tecnologias educacionais recentes que definem padrões interoperáveis; estão nesta lista diversos padrões que serão analisados neste texto: IEEE LOM, IMS CP, SCORM, etc.

Atualmente, o projeto possui cinco ferramentas:

- Metadata and Content Package Editor (Reload Editor)
- SCORM Player
- Learning Design Editor
- Learning Design Player

A Figura 5 apresenta uma tela de edição dos metadados educacionais segundo o padrão LOM (note-se que é necessário escolher a opção LTSN Profile, conforme está indicado na ilustração). Como é usual nas ferramentas para este padrão, é apresentado um formulário ao usuário, no qual cada campo corresponde a uma propriedade do LOM.

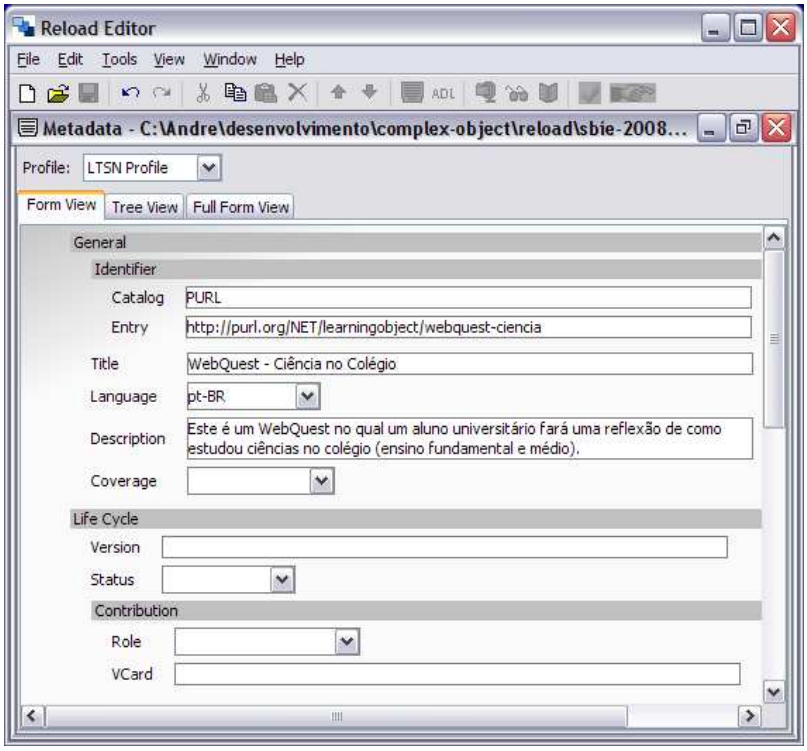


Figura 5. Tela de edição de metadados LOM do Reload Editor.

Neste exemplo apenas os seguintes campos foram preenchidos:

Campo	Descrição
Identifier **Catalog**: Padrão catalográfico escolhido. **Entry**: Identificação única conforme o padrão catalográfico.	Identificação única do objeto. Recomenda-se que não existam dois objetos dentro do contexto desejado com o mesmo identificador.
Title	Título do objeto que será lido pelo usuário.
Language	Linguagem em que os metadados estão descritos (pt-BR indica português do Brasil).
Description	Descrição do objeto que será lida pelo usuário.

Os seguintes valores foram inseridos nos campos

Campo	Descrição
Identifier	**Catalog**: PURL **Entry**: `http://purl.org/NET/learningobject/webquest-ciencia`
Title	WebQuest - Ciência no Colégio
Language	pt-BR
Description	Este é um WebQuest no qual um aluno universitário fará uma reflexão de como estudou ciências no colégio (ensino fundamental e médio).

O ideal é que o identificador criado seja único no mundo. Mas como conseguir isto?

Uma alternativa amplamente adotada é o uso de um URI – *Uniform Resource Identifier*, que estabelece um meio de identificar qualquer recurso de forma única dentro da Web. O tipo mais popular de URI é um subconjunto denominado URL – *Uniform Resource Locator* –, que identifica seus recursos por meio de sua localização física na Internet; ele é usado, por exemplo, para: páginas na Web, endereços de e-mail, diretórios ou arquivos para FTP, etc. Por outro lado, o URI envolve um conjunto mais abrangente, em que se podem definir identificadores igualmente únicos na rede, mas que não estão associados diretamente à localização física do recurso. Por intermédio de serviços de nomes, o identificador é relacionado com sua localização física na rede. Este mecanismo preserva o identificador, mesmo que o recurso mude sua localização na rede ou que o recurso não esteja mais disponível na mesma.

Este é o caso do identificador do exemplo cujo prefixo (`http://purl.org/NET/learningobject`) foi criado em um servidor de nomes chamado PURL (`http://purl.org`). O uso do PURL está entre as recomendações do CanCore [Friesen et al. 2004]. O CanCore (`http://www.cancore.ca`) é uma iniciativa canadense cujo objetivo é potencializar a busca e localização de conteúdo educacional. Ele define um conjunto de recomendações (CanCore Guidelines) [Friesen et al. 2004] baseadas no IEEE LOM. Para maiores detalhes veja [Dourado et al. 2007].

O arquivo produzido em XML para este exemplo é:

```
<lom …>
  <general>
    <title>
      <langstring xml:lang="pt-BR">
          WebQuest - Ciência no Colégio
      </langstring>
    </title>
    <catalogentry>
      <catalog>PURL</catalog>
      <entry>
        <langstring xml:lang="pt-BR">
          http://purl.org/NET/learningobject/webquest-ciencia
        </langstring>
      </entry>
    </catalogentry>
    <language>pt-BR</language>
    <description>
      <langstring xml:lang="en">
          Este é um WebQuest no qual um aluno universitário fará uma reflexão
          de como estudou ciências no colégio (ensino fundamental e médio).
      </langstring>
    </description>
  </general>
</lom>
```

Note-se que o mapeamento para XML usualmente ocorre da seguinte maneira: para um campo de nome “catalog” cujo valor é “PURL”, o mapeamento resulta em:

```
<catalog>PURL</catalog>
```

Quando o campo deve ser lido pelo usuário, usa-se o marcador `<langstring>` com um atributo que indica em que língua está escrito o texto (isso permite criar textos multilíngües).

Apesar das evidentes vantagens desse mapeamento, o XML foi concebido como um padrão para definir uma estrutura e um formato universais de representação de dados, ou seja, trata-se de um padrão de interoperabilidade sintática. O padrão RDF –

Resource Description Framework – da Web Semântica atua de forma complementar ao XML, proporcionando interoperabilidade na forma como estes dados são interpretados pelas aplicações (interoperabilidade semântica). O RDF define um modelo e uma linguagem para a representação homogênea de informações associadas a recursos que podem ser identificados por meio da Web [Manola 2003]. Tal representação é propícia para 'consumo por computadores' e atua diretamente sobre problemas de interoperabilidade.

O grupo de trabalho 12 (WG12) do LTSC indica a necessidade de uma futura compatibilidade do LOM com os padrões da Web Semântica, como está registrado na seção "LOM Next" do grupo (*http://www.ieeeltsc.org/working-groups/wg12LOM/LOMnext/*). Adicionalmente, Nilsson [Nilsson et al. 2003] apresenta uma proposta de mapeamento do LOM para RDF.

3.2. Empacotamento

Como pode ser observado nos Exemplos 2 e 3 de OA, e especialmente na Figura 4, um objeto pode ser constituído de um conjunto diversificado de artefatos digitais, relacionados entre si de diversas maneiras. Isso conduz à questão: como organizar tal conteúdo em uma só unidade para facilitar sua distribuição e reuso? Esta é a questão central tratada pelos padrões de empacotamento.

Mediante o processo de empacotamento, um OA (com seus metadados associados) é armazenado dentro de um pacote. Um pacote é uma estrutura responsável pela agregação de múltiplos artefatos digitais em um arquivo único, preservando sua organização.

<u>IMS CP</u>

O IMS CP (IMS Content Package) é um modelo para empacotamento de OAs definido pelo IMS *Global Learning Consortium* [Smythe and Jackl 2004].

A estrutura de um pacote de IMS CP é definida conforme a Figura 6. Na figura, a estrutura externa, indicada como Package, consiste em um arquivo compactado em formato ZIP. Dentro do ZIP, está um arquivo especial denominado manifesto (*manifest*) com quatro subdivisões. Este arquivo será comentado adiante. Além do manifesto, ficam dentro do ZIP todos os demais artefatos que compõem o OA.

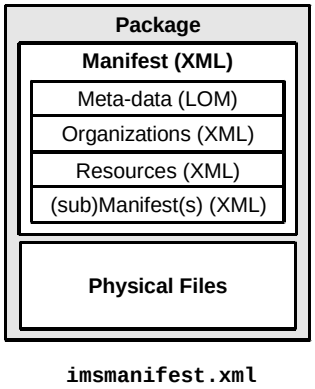


Figura 6. Estrutura do pacote IMS CP [Smythe and Jackl 2004].

Para se criar um pacote IMS CP pode ser usado o mesmo Reload Editor da subseção anterior. Na Figura 7 é apresentada a tela de edição do Reload Editor durante o processo de criação de um pacote IMS CP para o Exemplo 3 de OA.

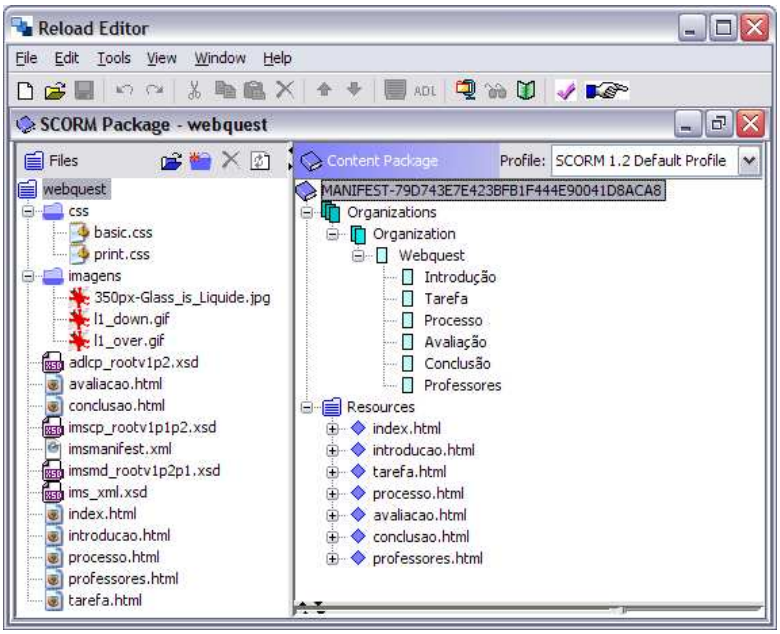


Figura 7. Reload Editor utilizado para criar um pacote IMS CP.

À esquerda da figura aparece a organização dos artefatos dentro do arquivo ZIP, à direita é detalhado o arquivo de manifesto. O destaque para o arquivo de manifesto ressalta a sua importância como estrutura complementar para a organização dos artefatos.

O formato ZIP não apenas reúne todos os artefatos em um único arquivo, como é capaz de reproduzir a estrutura hierárquica dos diretórios que compõem o objeto. A Figura 8 apresenta o pacote IMS CP da Figura 7 aberto por um programa de manipulação de arquivos ZIP. Como é possível observar, o formato IMS CP utiliza o formato ZIP sem modificações. Todas as informações adicionais relativas aos artefatos se concentram no arquivo de manifesto, que aparece na raiz do arquivo compacto (`imsmanifest.xml`).

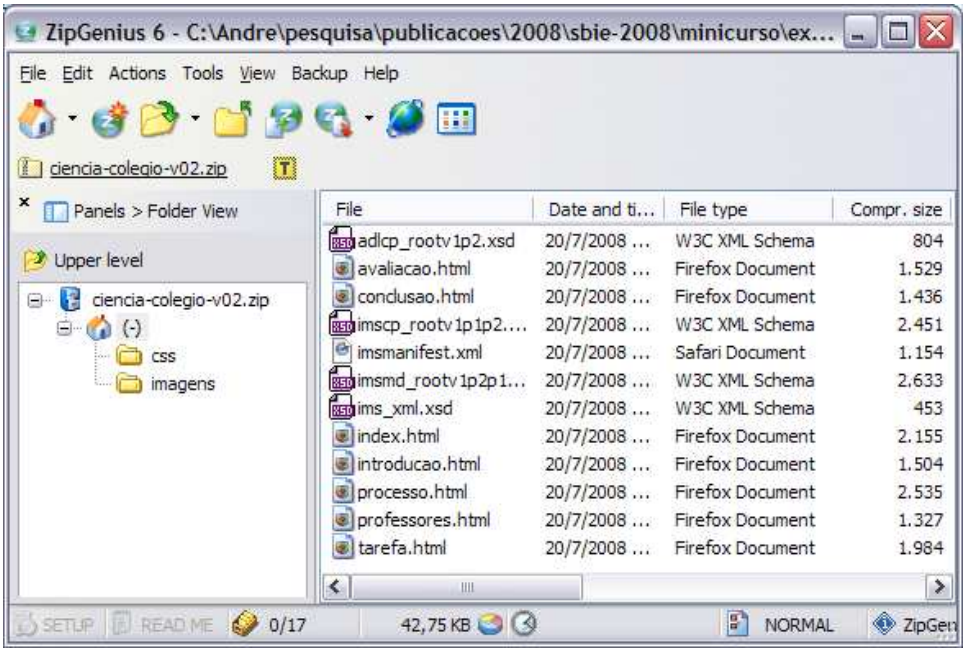


Figura 8. Pacote IMS CP aberto em um programa para formato ZIP.

No padrão IMS CP o manifesto é dividido nas seguintes seções (Figura 6):

- **Meta-Data** – metadados educacionais conforme o padrão LOM, que descrevem o objeto como um todo.
- **Organização** – define uma estrutura organizacional hierárquica que funciona como um índice de tópicos e subtópicos associados ao conteúdo educacional.
- **Recursos** – contêm referências para os artefatos que estão armazenados no ZIP e mapeiam as dependências entre eles.
- **Submanifestos** – esta seção é opcional, e contém manifestos subordinados quando há pacotes dentro de pacotes.

A seguir, apresenta-se outro exemplo prático de aplicação do IMS CP.

Considere um professor de biologia que está navegando pela Web, pesquisando conteúdo para a sua próxima aula sobre Células. Durante a pesquisa encontra o site MIT OpenCourseWare[1], no qual estão disponíveis recursos gratuitos para estudantes e professores. Ao acessar o site, ele escolhe sua área de interesse – Biologia. O link conduz para a página ilustrada na Figura 9.

[1] Massachussets Institute of Technology – OpenCourseWare - http://ocw.mit.edu/index.html

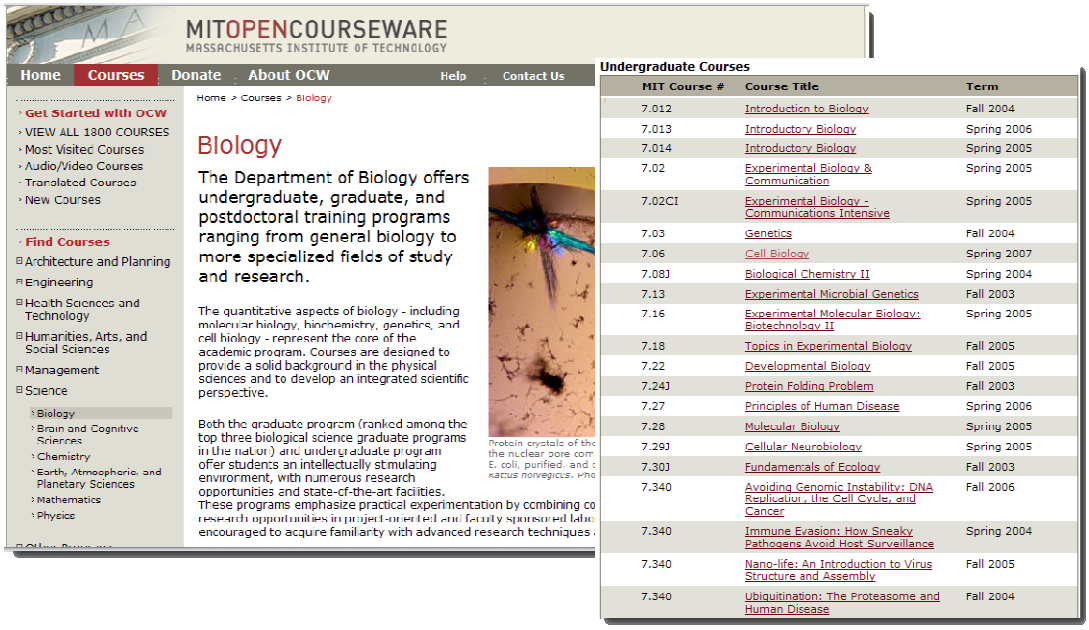


Figura 9. Cursos de Biologia do MIT OpenCouseWare.

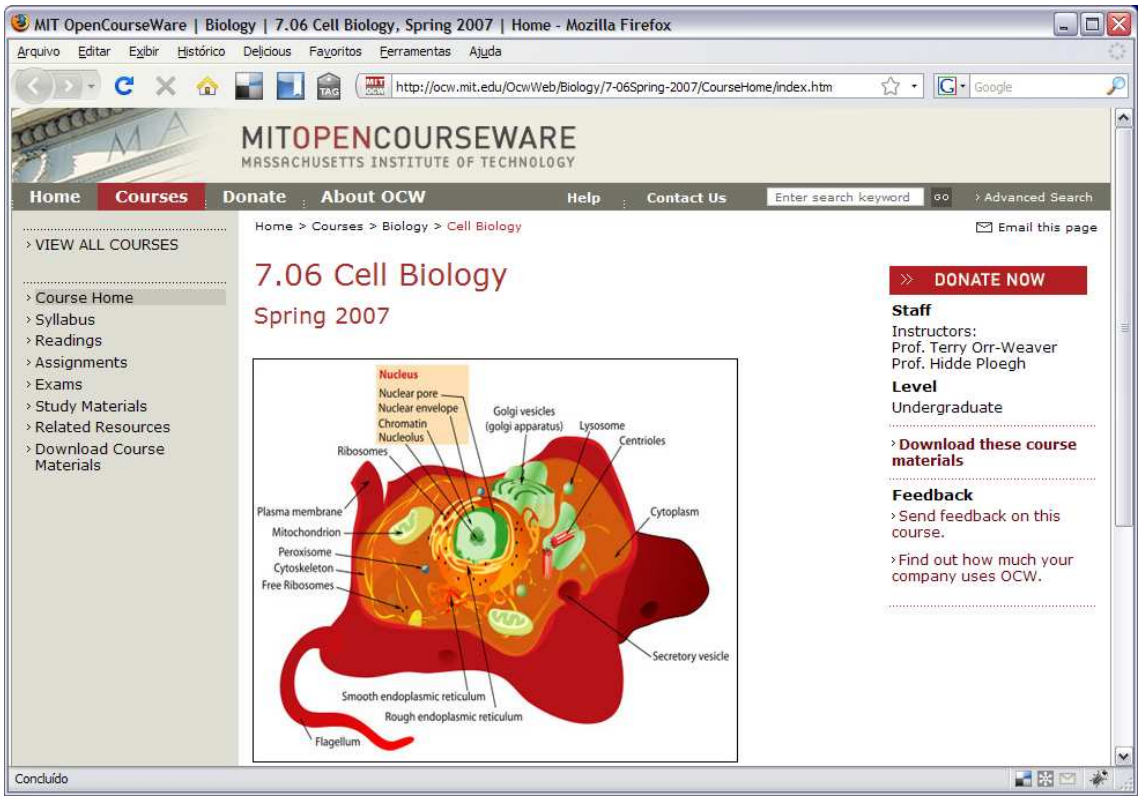


Figura 10. Página da disciplina 7.06 Cell Biology.

Em seguida, o professor seleciona um dos cursos disponíveis na lista apresentada. Considere-se que ele selecionou o curso de Biologia Celular. Será apresentada a página da Figura 10. Na figura à esquerda é mostrado o padrão de organização de um curso no MIT OpenCourseWare, que é padronizado para todos os cursos:

- **Course Home** – faz uma apresentação inicial, informando uma breve descrição do curso.
- **Syllabus** – sumário da programação do curso.
- **Calendar** – calendário do curso.
- **Readings** – o assunto Biologia Celular é apresentado em tópicos e, para cada um destes, são indicadas referências relevantes.
- **Lecture Notes** – notas para as aulas.
- **Assignments** – são disponibilizados alguns exercícios para serem executados.
- **Download this Course** – opção de fazer download do curso na forma de um pacote IMS CP.

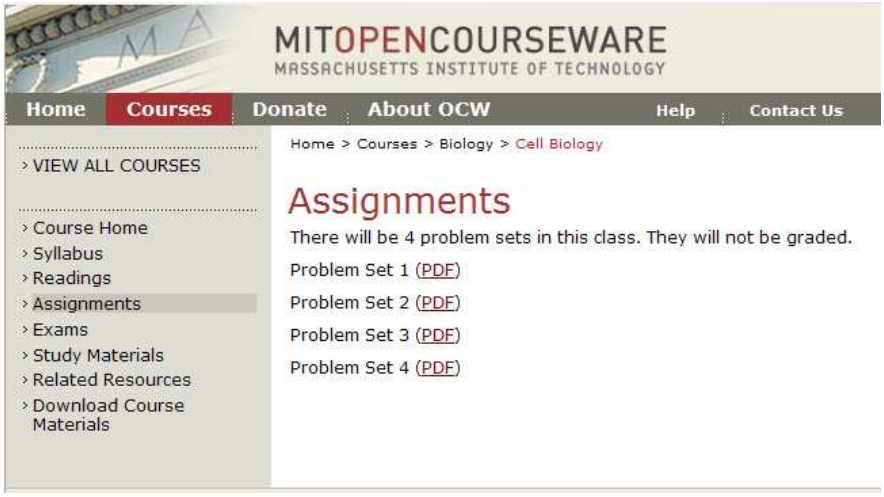


Figura 11. Seção de Tarefas do curso de Biologia Celular – MIT OCW.

Pode-se constatar que o conteúdo de um curso é formado por um conjunto diversificado de artefatos inter-relacionados: além das páginas Web, com suas figuras, estilos e scripts. Há ainda material associado em outros formatos – veja ainda que a Figura 11, à direita, mostra a seção "Assignments" (Tarefas), na qual o professor disponibilizou alguns arquivos PDF associados a uma determinada página Web.

Esse conteúdo pode ser caracterizado como um OA**.** Assim, para que o conteúdo possa ser compartilhado e reusado de forma consistente, é necessário que todos os itens que fazem parte da página estejam organizados em uma estrutura que mostre a dependência entre estes, bem como a sua organização. Por isso se usou o IMS CP. Conforme foi feito no exemplo anterior, a Figura 12 apresenta o Reload Editor editando o pacote do MIT OCW.

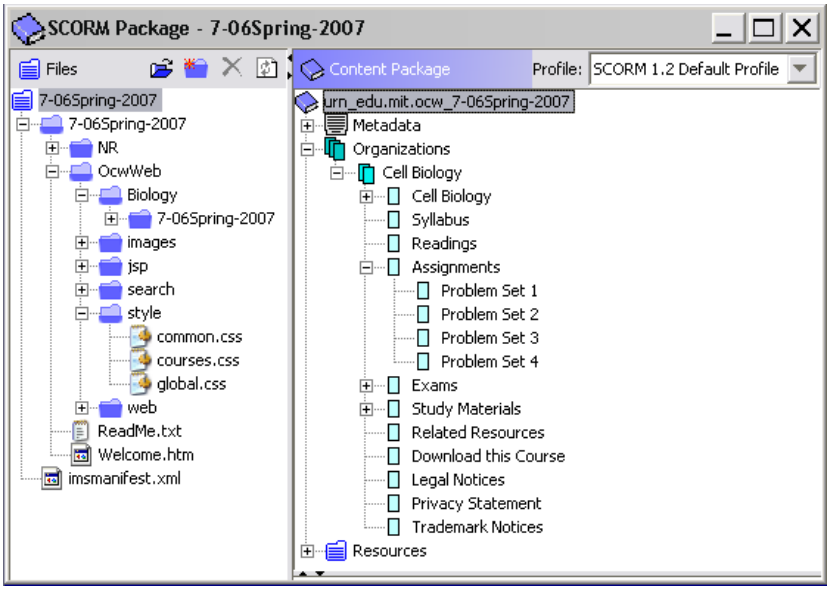


Figura 12. Pacote IMS CP do MIT OCW aberto no Reload Editor.

3.3. Comunicação com Sistemas de Gerenciamento de Cursos

Considerando-se que os OAs podem ser usados em atividades de aprendizagem gerenciadas por sistemas computacionais, são necessários padrões para que os objetos se comuniquem com o ambiente.

AICC CMI *Guidelines*

O *Aviation Industry CBT Committee* – AICC (`http://aicc.org`) definiu um padrão para a comunicação de módulos educacionais com ambientes de aprendizagem baseados na Web [McDonald et al. 2004]. Este padrão define API baseada em ECMAScript. Através dessa API, um OA pode solicitar serviços e dados do ambiente de aprendizagem e vice-versa. ECMAScript pode ser compreendido como um padrão que se propõe a padronizar e unificar as implementações da linguagem JavaScript [ECMA 1999] e, por esse motivo, utilizaremos daqui em adiante o termo JavaScript para nos referirmos a ECMAScript.

SCORM RTE

O padrão integrador SCORM (que será detalhado adiante) derivou o padrão AICC para a definição do SCORM RTE (*Run Time Environment*) [ADL 2006]. O RTE estabelece uma API de comunicação entre o SCORM SCO (*Sharable Content Object*) e o Sistema de Gerenciamento de Cursos (SGC). Um SCO é a menor unidade de um pacote SCORM capaz de interagir com o SGC através da API JavaScript.

O RTE parte do pressuposto de que o SCO suporta rotinas em JavaScript (e.g., um módulo Web, tal como uma página HTML). Se por um lado essa característica permite simplificar a infra-estrutura de interação entre o módulo e o ambiente, dado que o próprio navegador Web oferece suporte a JavaScript, por outro, ela limita as opções de linguagem. Essa restrição segue um pressuposto do padrão SCORM de que seus objetos devem ser construídos para serem executados na Web.

No diagrama da Figura 13 está ilustrada a interação entre um SCO SCORM e um SGC, especificamente o Moodle (que será apresentado mais adiante). Como mostra a figura, um SCO implementa uma rotina em JavaScript, que se comunicará com um módulo provido pelo Moodle e carregado diretamente no navegador (API SCORM RTE). Este módulo tem a responsabilidade de mediar a comunicação entre o SCO no cliente e o Moodle no servidor. O Moodle, por outro lado, possui um módulo específico para apresentar e fornecer suporte a pacotes SCORM.

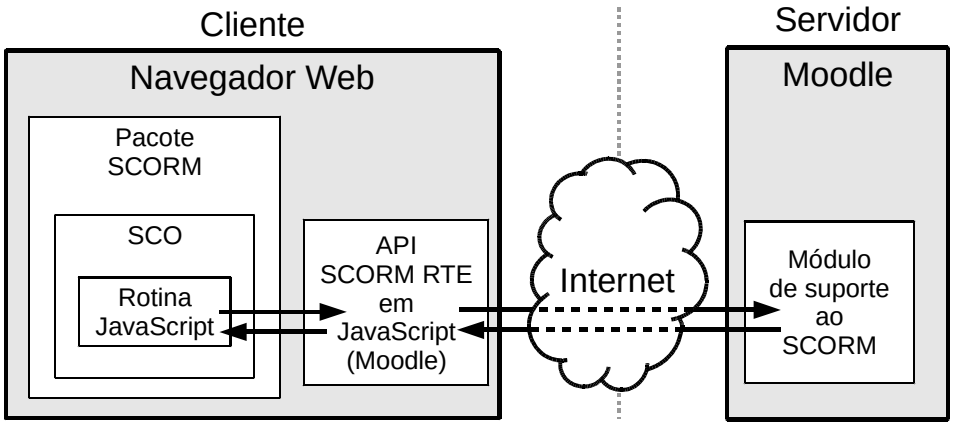


Figura 13. Diagrama de interação entre um SCO SCORM e o Moodle.

Na Figura 14 é apresentado um diagrama com a síntese da API, conforme especificação do SCORM RTE [ADL 2006]. Como mostra a figura, a API define funções para que o SCO: comunique ao ambiente quando começa e termina uma seção de comunicação (`Initialize()` e `Terminate()`); envie e receba valores (`GetValue()` e `SetValue()`); solicite o fechamento de uma transação com persistência de dados (`Commit()`); diagnostique e gerencie erros.

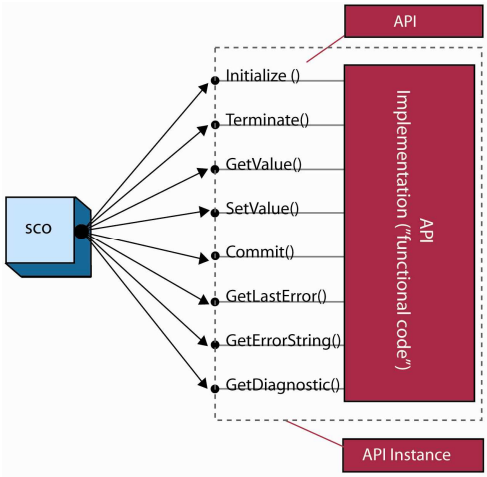


Figura 14. Síntese da API JavaScript do SCORM RTE [ADL 2006].

As funções `GetValue` e `SetValue` fazem referência a um conjunto pré-definido de variáveis do SCORM/AICC. Para fins de ilustração, apresentam-se a seguir quatro das múltiplas variáveis disponíveis (veja lista completa em [ADL 2006]):

`cmi.completion_status`	Indica se o estudante completou suas atividades no SCO.
`cmi.learner_name`	Nome do estudante.
`cmi.score`	Desempenho do estudante no desenvolvimento de alguma atividade no SCO.
`cmi.session_time`	Indica quanto tempo o estudante gastou na seção corrente do SCO.

Como pode ser observado, através destas variáveis o SCO pode obter dados do estudante (e.g., nome) e pode informar ao SGC dados de desempenho desse estudante. Dessa maneira o SGC pode acompanhar e computar o status de cada estudante no uso do SCO.

3.4. Integração

Com tantos padrões disponíveis surge a necessidade de integração. Este é o propósito do SCORM.

SCORM

O SCORM – *Sharable Content Object Reference Model* [ADL 2006] é uma iniciativa da *Advanced Distributed Learning* – ADL (`http://www.adlnet.gov`) que unificou o IEEE LOM com o IMS CP e a especificação da AICC para interfaces e regras. O resultado é uma infra-estrutura com suporte para reusabilidade, interoperabilidade, localização e adaptação. O padrão integrado reúne:

- Metadados com uma extensão e adaptação do IEEE LOM.
- Empacotamento com uma extensão e adaptação do IMS CP.
- Comunicação com uma extensão e adaptação do AICC.

As adaptações são importantes, principalmente para combinar e compatibilizar os padrões entre si. A iniciativa ADL distribui um "Conformance Test Suite" para garantir que novos produtos sigam o modelo de referência.

4. Ferramentas

Anteriormente foi apresentado o conjunto de programas do Reload Project, que consiste em um poderoso ferramental genérico para trabalhar nos mais diversos aspectos relacionados aos padrões abertos de OAs. Nesta seção apresentaremos outras ferramentas relacionadas à produção e uso de OAs, mais especificamente o SGC Moodle e ferramentas de autoria especializadas.

4.1. Objetos de Aprendizagem nos Sistemas para Gerenciamento de Cursos

Vamos retomar o exemplo da do MIT OCW para ilustrar o uso de OAs dentro de um Sistema para Gerenciamento de Cursos (SGC). Considere que, depois de fazer download do pacote, o professor do nosso exemplo deseja disponibilizá-la em um SGC, no nosso exemplo utilizaremos o Moodle.

O MOODLE – **M**odular **O**bject **O**riented **D**istance **LE**arning (http://moodle.org) – é um software de fonte aberta (Open Source Software), o que significa que se pode

instalar, usar, modificar e mesmo distribuir o programa (nos termos da GNU General Public Licence). O Moodle pode ser usado, sem modificações, em Unix, Linux, Windows, Mac OS e outros sistemas de suportem PHP. Ele está disponível em 40 idiomas.

Como um SGC, o Moodle disponibiliza ferramentas de produção, comunicação e gerenciamento das interações dos usuários. Sua plataforma dispõe de várias ferramentas de comunicação, como o forum e o chat. Para produções coletivas possui a ferramenta WIKI, enquetes, glossários e base de dados que suportam vários usuários ao mesmo tempo. Na perspectiva de gerenciamento, é possível visualizarde forma simples as entradas e as participações dos usuários. O Moodle se apresenta como um interessante ambiente virtual de aprendizagem, que pode auxiliar o professor na produção de cursos, grupos de estudos, ambientes de simulações, no sentido de promover e potencializar espaços-tempo de aprendizagem.

Vejamos o uso do pacote IMS CP vinculado ao Moodle. Para isso o professor irá:

1. Escolher a opção “Usar um pacote IMS CP” dentro do Moodle.
2. Importar o pacote ZIP contendo o objeto de aprendizagem e nomeá-lo.
3. Usar o pacote.

O resultado é ilustrado na Figura 15. Note que a estrutura definida originalmente pelo MIT OCW se replica na estrutura de organização do pacote, e novamente no Moodle.

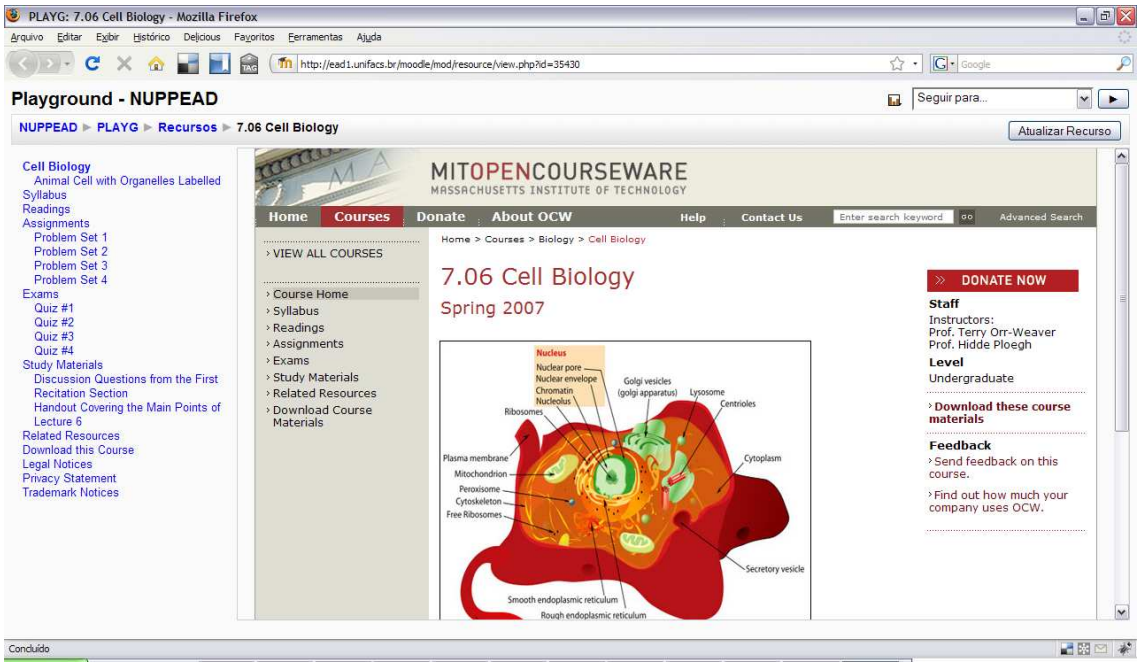


Figura 15. Pacote IMS CP do MIT OCW dentro do Moodle.

É importante ressaltar que a inserção de um pacote através da opção “Usar um pacote IMS CP” é usada para pacotes exclusivamente IMS CP. Pacotes em formato

SCORM devem usar a opção “SCORM/AICC” do Moodle, que ativará o suporte do Moodle à interface AICC.

4.2. eXe – eLearning XHTML editor

O eXe – eLearning XHTML editor (http://exelearning.org) – é um ambiente de autoria open-source que é capaz de exportar o conteúdo produzido para os formatos IMS CP e SCORM. A ferramenta é executada diretamente sobre o navegador Web, permitindo que o autor tenha uma visão bastante fiel do resultado durante a produção do conteúdo.

As produções em eXe são feitas diretamente sobre páginas HTML. O autor insere unidades chamadas iDevices. Cada iDevice tem uma funcionalidade diferente: apresentação de conteúdo, questionário, atividade etc. A Figura 16 ilustra uma aplicação feita em eXe utilizando o iDevice de questionário. Essa aplicação foi exportada para o formato SCORM e inserida no Moodle.

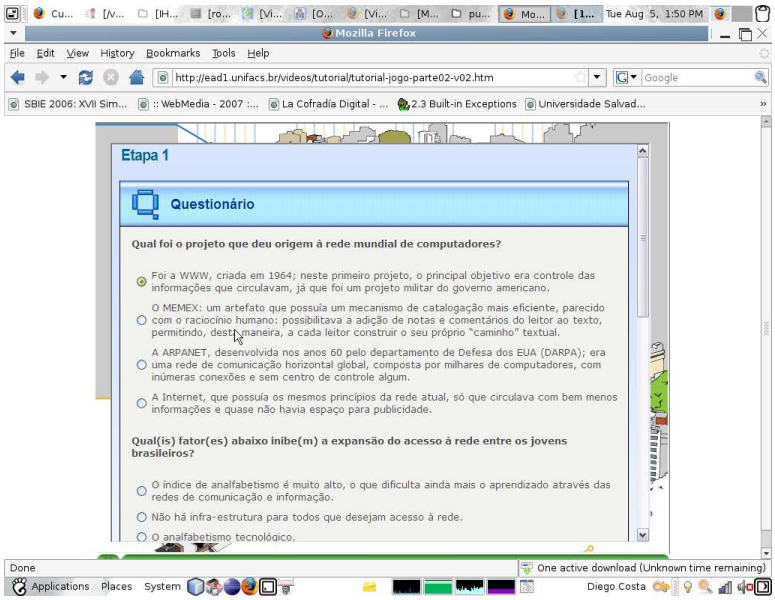


Figura 16. Questionário produzido em eXe e executado no Moodle.

Questionários exportados pelo eXe são capazes de interagir com o SGC através do padrão SCORM/AICC. Desse modo, o desempenho do estudante no questionário pode ser enviado a um SGC – tal como o Moodle – e computado pelo mesmo na avaliação do estudante.

4.3. CourseLab

Tal como o eXe, o CourseLab (http://www.courselab.com) é uma ferramenta de autoria que é capaz de exportar seu conteúdo para SCORM. O CourseLab é de uso gratuito.

O CourseLab utiliza a metáfora de slides para sistematizar o conteúdo produzido. Na Figura 17 é apresentada uma tela do CourseLab durante o processo de produção de um WebQuest (vide exemplo de OA 3).

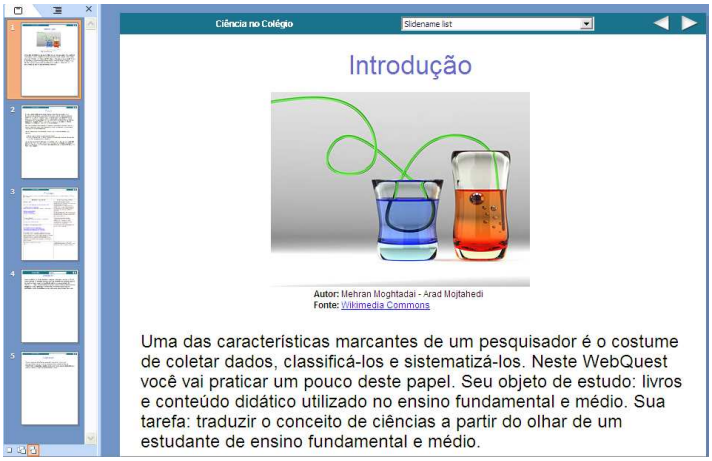


Figura 17. Tela do CourseLab na produção de um WebQuest.

Este software dispõe de uma rica galeria de recursos prontos, bem como suporte à construção de animações. Tal como acontece no eXe, além de dispor de múltiplas formas de criação de conteúdo, o CourseLab também pode ser usado para a criação de atividades e questionários, que enviam os resultados do desempenho do estudante para um SGC através da interface SCORM/AICC.

5. Considerações Finais

Este texto atua de forma complementar ao nosso minicurso anterior [Santanchè 2007], enriquecendo-o em três aspectos: (i) detalhando a interface de interação entre OAs e SGCs através do padrão AICC; (ii) descrevendo novos exemplos práticos; (iii) apresentando ferramentas de autoria de uso livre.

Este texto pretende contribuir principalmente na difusão dos benefícios da adoção de padrões abertos, para que o sonho dos OAs se torne uma realidade, através da construção coletiva de uma extensa biblioteca mundial de objetos.

Agradecimentos

Este trabalho foi parcialmente financiado pelos projetos LabMultflex (FAPESB-Infra 2006), MediaBank (HP Digital Publishing), WebMaps II (CNPq) e Bio-CORE (CNPq).

Referências

ADL (2006). Sharable Content Object Reference Model (SCORM) 2004 – 3rd Edition – Overview. `http://www.adlnet.gov/downloads/DownloadPage.aspx?ID=237`, accessed on 10/2007.

Dodge, B (1995). WebQuests: a technique for internet – based learning. San Diego: The Distance Educator, pages 10-13.

Dodge, B (2001). FOCUS: five rules for writing a great webquest. Learning & Leading With Technology, pages 6-9.

Dourado, P., Ferreira, P., and Santanchè, A. (2007). Representação Unificada de Objetos Digitais Complexos: confrontando o RAS com o IMS CP. In *Proc. of III Workshop on Digital Libraries*.

ECMA International (1999). ECMAScript Language Specification – Standard ECMA-262. http://www.ecma-international.org/publications/standards/Ecma-262.htm, accessed on 11/2004.

Friesen, N., Fisher, S., and Roberts, A. (2004). Cancore guidelines – for the implementation of learning object metadata (ieee 1484.12.1-2002) version 2.0. http://www.cancore.ca/en/guidelines.html, accessed on 10/2007.

IEEE L.T.S.C. (2000). Draft Standard for Learning Object Metadata – IEEE P1484.12/D4.0.

IEEE L.T.S.C. (2002). Draft Standard for Learning Object Metadata – IEEE 1484.12.1-2002. http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf, accessed on 09/2007.

L'Allier, J. J. (1997). Frame of Reference: NETg's Map to Its Products, Their Structures and Core Beliefs. http://www.im.com.tr/framerefer.htm, accessed on 10/2007.

McDonald, W. A., Hyde, J., and Montgomery, A. (2004). CMI Guidelines for Interoperability AICC. www.aicc.org/docs/tech/cmi001v4.pdf, accessed on 10/2007.

Nilsson, M., Johnston, P., Naeve, A., and Powell, A. (2006). *The Future of Learning Object Metadata Interoperability*, pages 255–314. Informing Science Press.

Nilsson, M., Palmér, M., and Brase, J. (2003). The LOM RDF binding – principles and implementation. In *Proc. of the Third Annual ARIADNE conference*.

Santanchè, A. ; Lago, A. ; Dourado, P. ; Ferreira, P. (2007). Objetos Digitais Complexos na Educação e os Objetos de Aprendizagem. In *Anais do XVIII Simpósio Brasileiro de Informática na Educação* -- Mini Cursos.

Smythe, C. and Jackl, A. (2004). IMS Content Packaging Information Model – Version 1.1.4 Final Specification. Specification, IMS Global Learning Consortium, Inc. www.imsglobal.org/content/packaging/cpv1p1p4/imscp_infov1p1p4.html, accessed on 08/2007.

Sosteric, M. and Hesemeier, S. (2002). When is a learning object not an object: A first step towards a theory of learning objects. *The International Review of Research in Open and Distance Learning*, 3(2).

von Brevern, H. (2004). Cognitive and logical rationales for e-learning objects. *Journal of Educational Technology & Society*, 7(4):2–25.

Wiley, D. A. (2000). Connecting learning objects to instructional design theory: A definition, *a metaphor, and a taxonomy*. Online version: http://reusability.org/read/chapters/wiley.doc.